So Near and Yet So Far: Distance-Bounding Attacks in Wireless Networks

Jolyon Clulow, Gerhard P. Hancke, Markus G. Kuhn, Tyler Moore

Computer Laboratory, University of Cambridge
15 JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom
firstname.lastname@cl.cam.ac.uk

Abstract. Distance-bounding protocols aim to prevent an adversary from pretending that two parties are physically closer than they really are. We show that proposed distance-bounding protocols of Hu, Perrig and Johnson (2003), Sastry, Shankar and Wagner (2003), and Čapkun and Hubaux (2005, 2006) are vulnerable to a guessing attack where the malicious prover preemptively transmits guessed values for a number of response bits. We also show that communication channels not optimized for minimal latency imperil the security of distance-bounding protocols. The attacker can exploit this to appear closer himself or to perform a relaying attack against other nodes. We describe attack strategies to achieve this, including optimizing the communication protocol stack, taking early decisions as to the value of received bits and modifying the waveform of transmitted bits. We consider applying distance-bounding protocols to constrained devices and evaluate existing proposals for distance bounding in ad hoc networks.

1 Introduction

Distance-bounding protocols are specialized authentication protocols that determine an upper bound for the physical distance between two communicating parties [1]. They aim to prevent attackers from pretending that the prover is closer to the verifier than is actually the case. Distance-bounding protocols have been suggested for application in access control tokens (e.g., contact-less smartcards that open doors), to prevent *relaying* attacks where a local attacker relays a challenge to a distant token that returns a valid response. Distance bounding is an integral aspect of many secure localization or positioning proposals where the location of nodes is inferred from their communication [2].

Such knowledge is useful for mapping the topology of the network and for geographically aware routing algorithms [3]. Therefore, distance bounding has also been proposed as a protective measure for wireless networks, where relaying attacks (in this context also known as *wormhole* attacks) could be used to circumvent key establishment and routing protocols [4,5,6] if an adversary tunnels messages across the network using a low latency, out-of-band channel [5,7]. This emulates nodes at either end of the wormhole being closer than they actually are.

L. Buttyan, V. Gligor, and D. Westhoff (Eds.): ESAS 2006, LNCS 4357, pp. 83–97, 2006.

Distance bounding provides a mechanism for a node to determine whether another node is a genuine neighbor, that is, physically located within its communication radius. Neighbors are in a position of trust and integral to the correct operation of a wireless network. Confidentiality and authentication are achieved using keys shared between neighbors and it is through neighbors that nodes communicate with the rest of the network. Neighboring nodes also serve as intermediaries when path keys are established between two nodes that do not share a pre-assigned key. Finally, it is the neighbors of a node that can best detect when it is compromised and that are typically used in revocation, reputation or voting schemes. Masquerading as a neighbor therefore provides the basis for mounting attacks on routing, key establishment and revocation.

We consider the secure implementation of distance-bounding protocols in ad hoc, wireless networks. We observe that typical transmission formats and modulation techniques introduce latencies, which the adversary can reduce substantially, allowing him to appear closer to the verifier than his actual position. Similarly, the symbol detection mechanism of a receiver can be optimized to provide an early indication of received bits. This provides a "head start" but increases the possibility of transmission errors. It is also possible for an adversary to extract timing advantage from bit transmission by delaying to the last possible moment and then broadcasting at a significantly higher power level. While this does create a different waveform, receivers that integrate the signal over the whole period and decode the symbol based on the area under the waveform will see the same outcome. These attack strategies highlight additional security-critical requirements that distance bounding implementations must meet.

Section 2 provides some background to distance-bounding protocols. We then discuss possible attacks on time-of-flight distance-bounding protocols and present general principles for secure distance bounding in Section 3. Section 4 reviews some proposals to apply distance-bounding techniques in ad hoc and sensor networks and comments on their security. The appendix relates our insights to existing sensor-mote technology.

2 Background

Distance and location measurement has countless applications, most notably in navigation and construction. In wireless networks, we aim to infer the location of potentially mobile devices using existing communication channels. This prompts consideration of distance bounding and secure localization protocols.

Secure location services provide relative or absolute location of nodes within the network [8,9]. This requires not only the ability to calculate distances or angles, but also collaboration between multiple nodes, including 'anchor' or base station nodes that provide trusted reference location information [2]. Secure location services can leverage the existence of multiple nodes or base stations to cross reference, repeat and verify measurements to defend against malicious behavior [10,11,12,13,14].

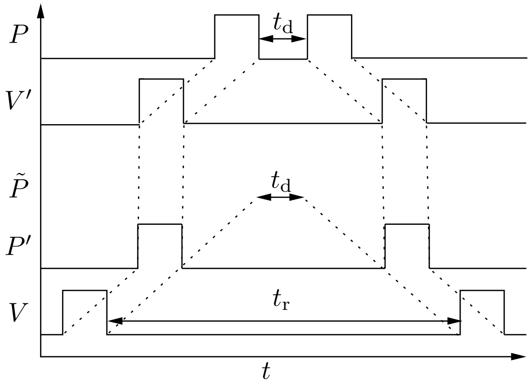

Fig. 1. Relay attack with slow medium: The vertical axis indicates node position. The attacker places a fake prover P' and verifier V' near the genuine verifier V and prover P, respectively. V' and P' communicate over a speed-of-light channel, while P and V use a slow speed-of-sound channel. A challenge issued by V is relayed by P' and V' much faster, and therefore received by P prematurely. The same may happen for the response. V measures a reduced round-trip time t_r and calculates, based on the assumed propagation speed and P's processing delay t_d, an artificially close position $\tilde{P}$ for P.

By contrast, distance bounding only involves two parties, a prover and a verifier, and allows the verifier to place an upper bound on the physical distance to the prover. Unlike secure location services, distance bounding relies exclusively on the protocol and communication medium to ensure security. Thus the requirements are more stringent.

Location-finding techniques generally use one of the following three basic methods:

- **Received Signal Strength (RSS):** Uses the inverse relationship between signal strength and distance to estimate the distance to other nodes [15].
- **Angle-of-Arrival (AoA):** Examines the directions of received signals to determine the locations of transmitters or receivers.
- **Time-of-Flight (ToF):** Measures elapsed time for a message exchange to estimate distance based on the communication medium's propagation speed.

The first two approaches are usually disqualified from security applications since attackers can easily alter received signal strength, by either amplifying or attenuating a signal, and angle-of-arrival, by reflecting or retransmitting from a different direction. This leaves only time-of-flight as a possible mechanism for secure location finding. Both radio frequency (RF) and ultrasound channels have been used in location systems. Since the propagation speed of sound is six orders of magnitude slower than light, the acoustic channel makes it easier to obtain high spatial resolution using simple hardware. However, ultrasound is vulnerable to a relay attack where messages are forwarded over a faster communication medium, as shown in Figure 1.

In contrast, the propagation speed of radio waves in air approaches the in-vacuum speed of light. Thus it resists simple relay attacks since information cannot propagate faster than this. The attacker can only make a node appear further away by blocking a legitimate node's communication and sending a delayed version to the intended receiver. While implementation on constrained devices can be a challenge, RF is already an established medium for mobile communication. So it is an ideal candidate for implementing distance-bounding systems.

2.1 Time-of-Flight Distance-Bounding Protocols

'Timed authentication protocols' are early, unsophisticated attempts to construct time-of-flight based distance-bounding protocols. The basic idea is to execute a challenge-response authentication protocol under a very tight time-out constraint. For example, a verifier V transmits a random n-bit nonce $N_V \in_{\mathrm{R}} \{0,1\}^n$ to the prover P, who replies with a message-authentication code $h_K(N_V)$, where h is a keyed pseudo-random function and K is a shared secret. Numerous protocols have been proposed using different constructions for pseudo-random functions keyed with shared secrets, public-key mechanisms, or trusted third parties. Examples in the literature include [5,16].

Conventional authentication protocols suffer from a common failing: it is not practical to implement the necessary time-out accurately enough over normal communications layers. The transmission time for full data packets and processing delays prevent such protocols from achieving the timing accuracy required.

In contrast, protocols specifically designed for distance-bounding applications do not transmit entire data packets. Rather, they operate at the bit level by recording individual bit-arrival times. We now review several such protocols.

Bit stream with timed reception: These protocols assume that both the verifier and the prover share a common, trusted, high-precision time base (e.g., secure GPS receivers). The verifier sends out random bits $C_1, C_2, \ldots, C_n$ at times $t_1, t_2, \ldots, t_n$ (where $t_i = t_0 + i \cdot t_{\mathrm{p}}$). The prover receives at its antenna input the bit values $C'_1, C'_2, \ldots, C'_n$ at times $t_1 + \Delta t, t_2 + \Delta t, \ldots, t_n + \Delta t$. It then replies with a message-authenticated data packet

$$\{t_0 + \Delta t, C'_1, C'_2, \ldots, C'_n\}_K.$$

The verifier checks the message-authentication code of this packet with the shared key K and verifies that $C_i = C'_i$ for at least $k > \frac{n}{2}$ different values $i \in \{1, \ldots, n\}$, where k and n are security parameters. Finally, the verifier checks whether $\Delta t \leq d/c$, where d is the upper bound for the distance and c is the speed of light. Setting $k < n$ allows for some transmission errors. (For brevity, we omit here technical details on how both sides agree *a priori* or *a posteriori* on $t_0 + \Delta t$.)

Duplex bit streams: In the absence of a common trusted clock, the class of protocols just outlined can be extended to transmit random data in both directions simultaneously [1]. The verifier sends C_i at $t_i = t_0 + i \cdot t_{\mathrm{p}}$ as before, which the prover again receives at times $t_i + \Delta t$, but now the prover also sends random bits R_i in the opposite direction at times $t_i + \Delta t$ (e.g., on a different radio frequency), which the verifier receives at times $t_i + 2\Delta t$ as R_i'. The prover finally transmits a message-authenticated data packet

$$\{C_1', C_2', \ldots, C_n', R_1, R_2, \ldots, R_n\}_K.$$

The verifier checks the message-authentication code with key K, then verifies that $C_i = C_i'$ and $R_i = R_i'$ for at least $k > \frac{n}{2}$ different values $i \in \{1, \ldots, n\}$, where k and n are security parameters, and finally checks whether $\Delta t \leq d/c$. Instead of authenticating for each received value C_i' the corresponding time, in this variant, the prover authenticates what it sent out in the other direction at the time of receiving C_i'.

In both protocols, the prover can easily cheat, either by lying about $t_0 + \Delta t$ or by sending R_i before receiving C_i'. Therefore, these protocols can only defend against third-party attackers that do not have access to the shared secret key K. Such cheating can be made more difficult if R_i is not simply an unpredictable random bit, but is calculated as a function of C_i'. It is important that the processing time is minimized to reduce the uncertainty of the distance-bounding process. Therefore, the function $g(i, C_i') \mapsto R_i$ must be easy to implement with only a few gate delays. Two such approaches have been described in the literature.

Bitwise XOR with pre-commitment: Both the verifier and prover first generate random bit strings $C = (C_1, C_2, \ldots, C_n)$ and $M = (M_1, M_2, \ldots, M_n)$, respectively. The prover commits to M (e.g., by transmitting a collision-resistant message authentication code $h_K(M)$). The verifier then sends one C_i after another, which the prover receives as C_i'. It then instantly replies with a bit $R_i = C_i' \oplus M_i$, which is calculated by XOR-ing each received challenge bit with the corresponding bit of M. Finally the prover reveals M and authenticates C'. The commitment on M is needed to prevent the prover from sending a random bit R_i early and then setting $M_i = C_i' \oplus R_i$ after receiving C_i'. Authenticating C' keeps attackers from sending fake C_i bits prematurely to the prover to learn bits of M_i for responding early to the verifier.

This construction first appeared in the Brands-Chaum protocol [1] and has inspired a number of variants [7,12,13]. As was pointed out in [17], this protocol can tolerate bit errors in the transmission of the C_i and R_i as long as the C' received and the M applied are afterwards transmitted over an error-corrected channel. The verifier can then accept the response if $R_i' = C_i \oplus M_i$ for at least k_1 bits i and $C_j' = C_j$ for at least k_2 bits j, where $k_1, k_2 > \frac{n}{2}$ and n are security parameters.

Pre-computed table lookup: The verifier generates a random bit string $C_1, C_2, \ldots, C_n$ and a nonce N_V that is sent to the prover. The prover responds with its nonce N_P. Both the prover and the verifier then use the pseudo-random function h and the secret key K in order to calculate two n-bit sequences R^0 and R^1:

$$(R_1^0, R_2^0, R_3^0, \ldots, R_n^0, R_1^1, R_2^1, R_3^1, \ldots, R_n^1) \ := \ h_K(N_V, N_P)$$

The prover's reply bit $R_i = R_i^{C_i'}$ to each C_i' received from the verifier is the result of a 1-bit table lookup in R^0 or R^1, selected by the received challenge bit C_i' (for $1 \leq i \leq n$). The verifier checks whether at least k of the n R_i' bits that it receives match its locally calculated $R_i^{C_i}$ values. The values $k > \frac{3}{4}n$ and n are security parameters. The Hancke-Kuhn protocol [17] presents this strategy, which has the advantage that no further data has to be exchanged once the rapid bit exchanges have taken place.

Accuracy The accuracy of the distance bound is influenced by the precision or resolution of the timing mechanism, properties of the communication channel including pulse width and bit period t_p, and processing delay t_d between receiving a challenge and sending the response.

Both the bitwise XOR with pre-commitment and pre-computed table lookup classes of protocols are designed to minimize the processing delay t_d. The former achieves this through the use of a fast operation (i.e., XOR) while the latter allows for pre-computation by the prover entirely before the time-critical challenge-response phase begins. In contrast, timed authentication protocols require the online generation of a signature or message authentication code during the timed period. Not only does this introduce an inaccuracy into the distance calculation but a malicious prover with high performance hardware can extract a time advantage by performing these operations faster. The effect is more pronounced and debilitating for constrained devices.

A single-bit exchange provides the highest time (and therefore distance) resolution, as it depends only on propagation time, pulse width and processing delay. Resolution also motivates the proposed use of ultra wideband or similar communications for distance bounding [18,19,20]. These are characterized by short pulse width and are already used in current location systems with resolution in the order of 30 cm [21]. Multiple timed message exchanges may appear inefficient but multiple measurements increase accuracy and confidence.

In contrast, some authors propose timing a single exchange of multi-bit challenge-response messages. For example, Čapkun and Hubaux describe essentially the Brands-Chaum protocol modified to a single message exchange [12,13]. In such systems, the choice of when to start and stop timing affects the resolution since it is now additionally dependent on the number of transmitted bits and the bit period, not just the pulse width. The greatest precision is obtained by timing from the transmission of the last bit of the challenge to the receipt of the first bit of the response. Care must be exercised to ensure that the first response bit depends on the last challenge bit. Čapkun and Hubaux achieve this by reversing the order of the response bits.

Bit errors Previously proposed protocols either fail in the event of a single bit error or require additional error correction overhead. This is not ideal in applications where communication errors are likely to occur and it is also vulnerable to a denial of service attack by an active adversary. We shall see later in Section 3 that resilience to noise is important requirement for security. Hancke and Kuhn [17] consider the impact of bit errors on distance-bounding protocols. The authors indicate how protocols can be modified to be resilient by specifying an error threshold.

3 Attacks on Time-of-Flight Protocols

3.1 Threat Model

Honest nodes adhere to their programmed strategy including algorithms for distance bounding. Malicious nodes can eavesdrop any message broadcast by an honest node. A malicious node can communicate with any other attacker-controlled node (via an out-of-band channel) as well as with honest nodes. Attacker-controlled nodes may modify any packet or transmission protocol, inserting or removing chosen identifiers, timestamps and location claims, message payloads and signatures. An attacker may have access to more sophisticated hardware and processing capabilities compared to that of normal devices.

We consider two attacks on distance-bounding protocols. A malicious prover can pretend to be closer to the verifier by responding faster than an honest node could. In a *relay attack*, malicious intermediaries seek to shorten the perceived distance between an honest prover and verifier. We do not consider here the case where a malicious prover colludes with another node that is located closer to the verifier, since a malicious prover can obviously always release all its secret keys to a colluder.

3.2 Guessing Attacks on Packet-Based Challenge-Response Protocols

Single-exchange challenge-response protocols with multi-bit messages are vulnerable to a guessing attack that enables a malicious prover to reduce the apparent distance to the verifier. The attack as applied to Čapkun-Hubaux [12,13] is shown in Figure 2. The key observation is that an adversary can guess the value for the last bit transmitted by the verifier and preemptively transmit a response. With probability $\frac{1}{2}$ the adversary guesses correctly and gains a timing advantage of up to twice the bit period. The advantage gained depends not on pulse width but on the bit period for the channel. So while n single-bit challenges reduce an attacker's chances of guessing the correct response to 2^{-n}, a single n-bit message can be shortened with probability $\frac{1}{2}$. An attacker can tailor his distance improvement according to his likelihood of success: he can shorten by $\Delta d \cdot l$ with probability 2^{-l}, where $\Delta d = 2t_{\mathrm{p}}c$ is the distance traversed during two bit periods. Furthermore, an attacker could exploit this even more if the protocol tolerates a specified threshold of errors. This weakness is present in the distance-bounding protocol proposals of Hu, Perrig and Johnson [5], Sastry, Shankar and Wagner [16], and Čapkun-Hubaux [12,13], and challenges the choice of a timed packet-based challenge-response exchange.

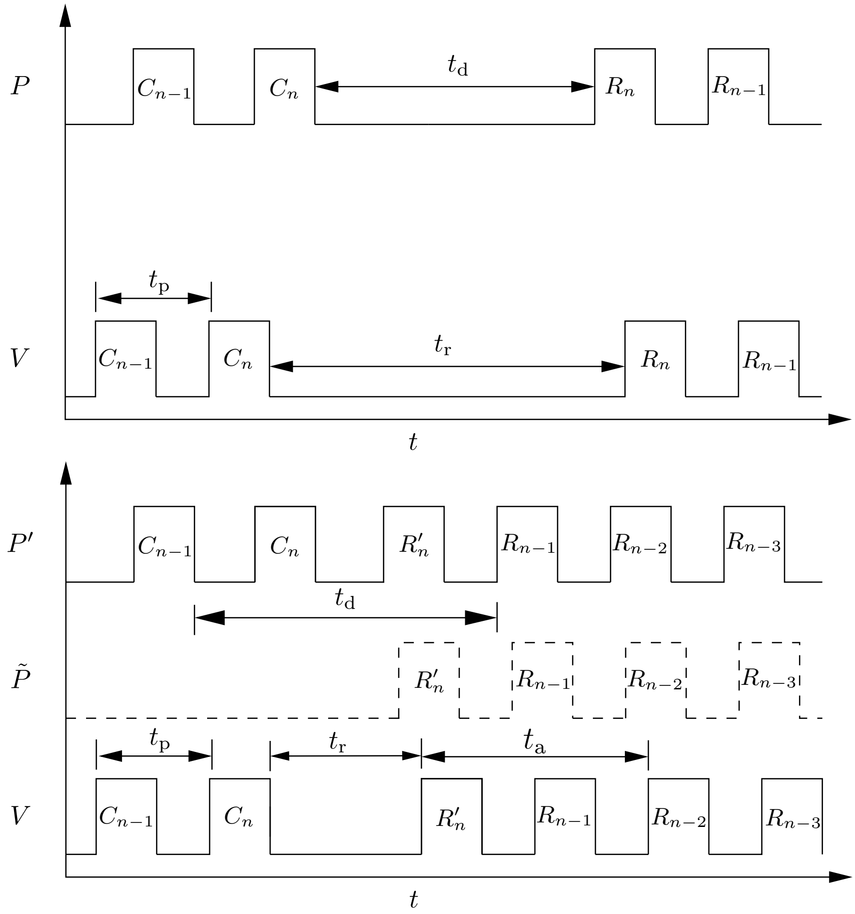

Fig. 2. The top figure shows normal operation of a single-exchange challenge-response protocol with the verifier calculating the distance bound from the measured round-trip time t_r. In the bottom figure, the malicious prover P' guesses the first response bit R'_n and transmits it after receiving challenge bit C_{n-1}. This gives the attacker enough time to calculate and respond with the correct response bit R_{n-1}, as well as all subsequent response bits. This yields timing advantage t_a equal to twice the bit period, so the verifier measures a shorter round-trip time t_r and perceives the prover at location $\tilde{P}$.

3.3 Exploiting Packet-Level Latencies

The security evaluation of a distance-bounding protocol must also consider ways in which an attacker could reduce any latency introduced by underlying communication layers. Most transmission formats and modulation techniques have been designed for robustness, ease of use, and power efficiency, rather than for minimizing transmission latency of individual data bits. Transmission software usually has to commit to an entire data block several bit times before the block's first data bit is actually transmitted. Likewise, the receiving software can only

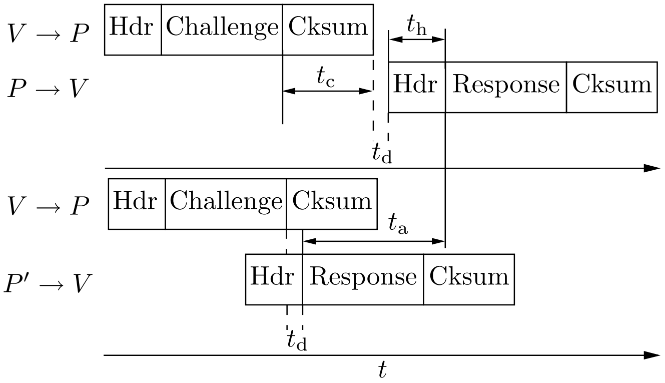

Fig. 3. If the verifier expects the prover to strictly adhere to the communication protocol, an attacker can gain time t_a equal to $t_c + t_h$. Time t_d is required to calculate the response once the entire challenge has been received. The attacker ignores the data trailer and starts calculating its response while preemptively transmitting the header of the return data.

access its content several bit times after the entire block has been received. In the simplest case, namely the asynchronous byte transmission scheme used on RS-232 lines, data blocks are just eight bits long and only a start and a stop bit are added as overhead. More commonly, data blocks comprise multiple bytes and are transmitted with synchronization preambles, headers with source and destination addresses and sequence numbers, as well as checksums and packet delimiters (HDLC, Ethernet, etc.). In the most sophisticated transmission schemes, error correcting encoders and decoders may add substantial further delays.

An attacker may not be restricted by the latencies imposed by regular implementations. It is often feasible to design special variant implementations of low-level communication standards, where the value of each data bit can be changed right up to the start of bit transmission, or where the receiving end is notified of each bit's value as it is decoded. An example of this attack is shown in Figure 3. (In practice, an attacker may have to replace a standard communications chip with an entirely software-based design, or an FPGA-based hardware/software codesign, to obtain such a specialized low-latency transceiver implementation economically.)

A possible overclocking attack is also worth noting. In many communication systems, the transmitter has control over the exact bit period t_p, and it is the responsibility of the receiver to recover the exact bit rate by extracting a clock signal embedded with the packet data (e.g., using Manchester coding). Recipients implement a phase-locked loop (PLL) circuit for this purpose, which must be able to tolerate certain deviations from the nominal frequency. An attacker who wants to appear closer may transmit at the maximum bit rate that the receiver's circuit still tolerates, leading to an earlier reception of the entire packet.

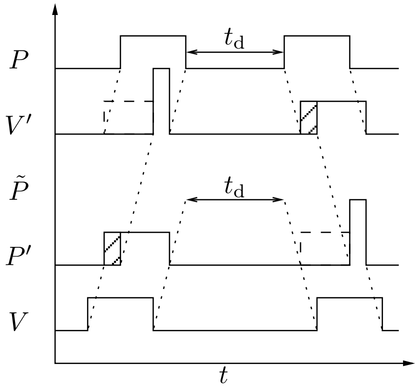

Fig. 4. In this variation of the relay attack the attacker gains time when P' estimates the value of the challenge bit from V early on in the bit period and V' transmits m-times the symbol amplitude to P in the final $\frac{1}{m}$-th of the bit period. The process is then repeated for the response bit, albeit with V' and P' swapping roles.

3.4 Deferred Bit Signalling

An attacker could also change a bit even after its transmission time has begun or act upon a received bit before its transmission has been completed. In simple modulation schemes, such as amplitude-shift keying (ASK) or frequency shift keying (FSK), each bit value is represented on the communication channel through the transmission of one of two different waveforms ("symbols"). Such a symbol might be one of two tones (FSK) or one of two amplitude levels (ASK). The receiver has to decide for each bit, in the presence of background noise, which symbol has most likely been transmitted. It does so by comparing the difference between the received waveform and the waveforms of the two candidate symbols, and integrates these differences over the entire duration of the symbol.

A regular transmitter makes the best use of its limited transmission power by spreading the energy available for each symbol as uniformly over the symbol's transmission time slot as possible (subject to constraints on transition times that bandwidth limitations bring). An adversary's modified implementation, however, may send no energy for $\frac{m-1}{m}$ of the time interval, and then may send the bit value during the final $\frac{1}{m}$-th of the available time, using a more powerful transmitter, with m-times higher amplitude than that used in a regular implementation. For the receiving end, which integrates the energy received over the entire symbol time, the result is the same, but the transmitter can delay committing to a bit's value by $\frac{m-1}{m}$ of a bit time. An example of this attack is shown in Figure 4.

3.5 Early Bit Detection

Likewise, an attacker may use a variant implementation of a receiver that does not wait for the decision of which bit has been received until all energy related to that bit has been received and integrated. If the attacker's receiver has an

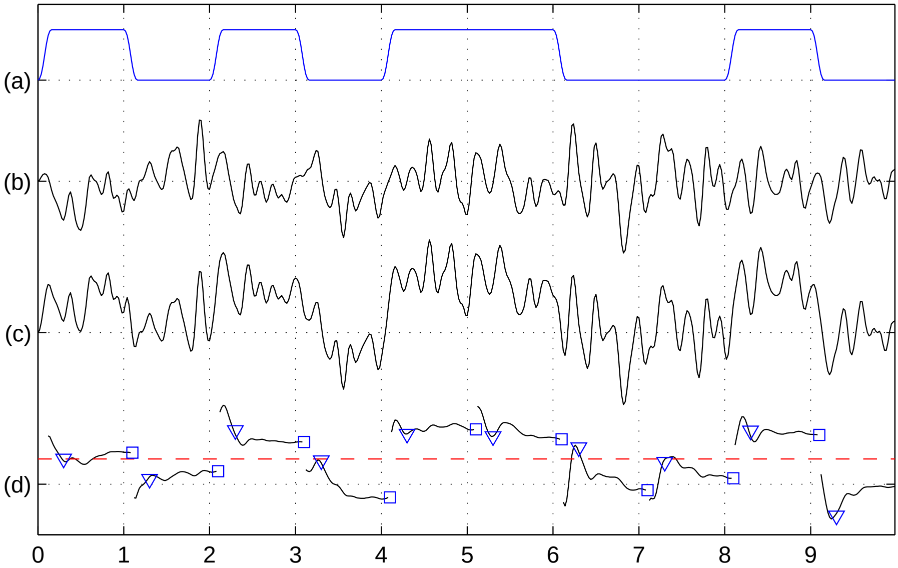

Fig. 5. Early decision decoder example, showing transmitted signal (a), added noise (b), and resulting received waveform (c). Curve (d) shows the result of averaging the received signal from the start of each bit. Squares mark the result of averaging the full bit length, and triangles the result of averaging only the first 20%. The dashed line represents the decision threshold (below: 0, above: 1). This early detection attempt leads only to a single bit error (bit 6) in this example.

m-times better signal-to-noise ratio than what a regular receiver really needs, then the attacker's receiver can terminate the integration already with $\frac{1}{m}$-th of the symbol's signal in (after about $\frac{1}{m}$ of the bit's transmission time), while still obtaining an acceptable bit error rate. This way, the attacker can save $\frac{m-1}{m}$ of the symbol's transmission time compared to using a regular receiver. The necessary m-times better signal-to-noise ratio could be achieved by reducing the distance to the receiver or with an antenna with better directional gain.

Figure 5 demonstrates the operation of a modified decoder in a receiver that was designed to provide an early decision for each bit compared to a conventional decoder. Waveform (a) is the output of the transmitter, which the receiver can see only along with an added noise signal (b), resulting in the received waveform (c). The receiver can achieve the best signal-to-noise ratio by processing (c) with a "matched filter", that is by multiplying the received waveform with the noise-free shape of a transmitted bit and integrating the result. In this example, the bits are represented by nearly rectangular pulses; therefore, the application of a matched filter is nearly equivalent to averaging the signal over the duration of one bit time. Waveform (d) in Figure 5 shows the result of averaging the received signal from the start of the current bit up to the current input value. The little squares show where this averaging process has integrated the whole length of the bit. At these points, the average output best represents the transmitted value and can be compared against the dashed threshold line to decide whether a 0 or 1 was received. To decide earlier, we must use an intermediate value of the average. The triangles on curve (d) show the value after only 1/5 of each bit has been received. These values are 4/5 of a bit time earlier available, but provide only 1/5 of the signal-to-noise amplitude ratio. This example shows a binary amplitude-shift-keying baseband signal in the interest of simplicity, but the principle can equally be applied to modulated complex symbols.

3.6 Principles for Secure Time-of-Flight Distance-Bounding Protocols

With all these attacks in mind, the designer of a distance-bounding protocol should optimize the choice of communication medium and transmission format according to the following principles:

- **Principle 1**: Use a communication medium with a propagation speed as close as possible to the physical limit for propagating information through space-time (the speed of light in vacuum). This excludes not only acoustic communication techniques, but also limits applicability of wires and optical fibers.
- **Principle 2**: Use a communication format in which only a single bit is transmitted and the recipient can instantly react on its reception. This excludes most traditional byte- or block-based communication formats, and in particular any form of error correction.
- **Principle 3**: Minimize the length of the symbol used to represent this single bit. In other words, output the energy that distinguishes the two possible transmitted bit values within as short a time as is feasible. This leaves the attacker little room to shorten this time interval further.
- **Principle 4**: The distance-bounding protocol should be designed to cope well with substantial bit error rates during the rapid single-bit exchange, because the previous criterion may limit the energy that can be spent on transmitting a single bit and conventional error correction is not applicable.

4 Existing Distance-Bounding Proposals

Secure Neighbor Detection The secure neighbor detection protocol proposed by Hu, Perrig and Johnson [5] is an instance of a timed authentication protocol where the elapsed time during the exchange of signed nonces infers a distance bound.

The protocol has significant processing overhead including hashing and then verifying and signing incoming and outgoing messages. While the authors discuss mechanisms for increasing the efficiency of the signing operations, the associated delay renders the bound inaccurate and unreliable. Furthermore, malicious nodes with higher performance components can extract a time advantage by performing these operations faster. The timing of only one multi-bit message exchange means the protocol is vulnerable to the guessing attack described in Section 3. We also note that the protocol is not robust in the presence of communication errors.

In-Location Verification Protocol Sastry, Shankar and Wagner [16] propose a timed authentication protocol to verify a prover's claimed physical location l within a circular region R centered on the verifier. The verifier issues a random challenge N to which the prover responds via a sound channel with $F_k(N)$ where F_k is a pseudo-random function. The verifier accepts this if $l \in R$ and the elapsed time is less than or equal to $d \cdot (c^{-1} + s^{-1})$ where c and s are the speed of radio waves and sound respectively and d is the distance.

Several authors have commented that this proposal is vulnerable due to its use of sound as a carrier, which contradicts Principle 1. We also criticize the use of a single challenge-response message exchange and a delay inducing pseudo-random function.

Čapkun-Hubaux Čapkun and Hubaux propose a distance-bounding protocol for use in secure positioning [12,13]. They modify the Brands-Chaum protocol by converting it into a single message exchange involving a multi-bit challenge-response.

Again, timing a single message exchange means the protocol is vulnerable to the guessing attack described in Section 3. We also note that the protocol is not robust in the presence of communication errors.

Mutually Authenticated Distance Bounding (MAD) Čapkun, Buttyán and Hubaux propose MAD [7], which modifies the Brands-Chaum protocol to allow both parties participating in the protocol to bound the distance to the other party simultaneously. This protocol does not suffer from the same bounding inaccuracies as those described above. Bits are exchanged over the radio channel; only single bits are transmitted rather than entire messages; no cryptographic operations are performed between timed exchanges. As with the Brands-Chaum protocol, a single bit error causes the protocol to fail; thus it is less suited for noisy channels.

5 Conclusion

In this paper, we have investigated the security of distance-bounding protocols for wireless networks. We have shown that time-of-flight techniques are vulnerable to several attacks: the round-trip time for a single timed multi-bit challenge-response can be reduced by guessing and preemptively transmitting response bits; communication layer protocol latencies can be avoided by the adversary; and time advantage can be extracted by modifying the transmission waveform and through the early detection of symbols. These attacks can be successfully applied to a number of existing proposals for use in ad hoc and sensor networks.

We propose a number of principles to adhere to when implementing distance-bounding systems. These restrict the choice of communication medium to speed-of-light channels, the communication format to single bit exchanges for timing, symbol length to narrow (ultra wideband) pulses, and protocols to error-tolerant versions. These restrictions increase the technical challenge of implementing secure distance bounding.

References

1. Brands, S., Chaum, D.: Distance-bounding protocols (extended abstract). In: EUROCRYPT. (1993) 344–359
2. Karl, H., Willig, A.: Protocols and Architectures for Wireless Sensor Networks. Wiley (2005)
3. Karp, B., Kung, H.T.: GPSR: greedy perimeter stateless routing for wireless networks. In: MOBICOM. (2000) 243–254
4. Hu, Y.C., Perrig, A., Johnson, D.B.: Packet leashes: A defense against wormhole attacks in wireless networks. In: INFOCOM. (2003)
5. Hu, Y.C., Perrig, A., Johnson, D.B.: Rushing attacks and defense in wireless ad hoc network routing protocols. [22] 30–40
6. Karlof, C., Wagner, D.: Secure routing in wireless sensor networks: attacks and countermeasures. Ad Hoc Networks **1**(2-3) (2003) 293–315
7. Čapkun, S., Buttyán, L., Hubaux, J.P.: SECTOR: secure tracking of node encounters in multi-hop wireless networks. In Setia, S., Swarup, V., eds.: SASN, ACM (2003) 21–32
8. Werb, J., Lanzl, C.: Designing a positioning system for finding things and people indoors. IEEE Spectrum **35**(9) (1998) 71–78
9. Bahl, P., Padmanabhan, V.: RADAR: An in-building RF-based user location and tracking system. In: Nineteenth Annual Joint Conference of the IEEE Computer and Communication Society, IEEE (2000) 775–784
10. Liu, D., Ning, P., Du, W.: Attack-resistant location estimation in sensor networks. In: IPSN, IEEE (2005) 99–106
11. Liu, D., Ning, P., Du, W.: Detecting malicious beacon nodes for secure location discovery in wireless sensor networks. In: ICDCS, IEEE Computer Society (2005) 609–619
12. Čapkun, S., Hubaux, J.P.: Secure positioning of wireless devices with application to sensor networks. In: INFOCOM. (2005)
13. Čapkun, S., Hubaux, J.P.: Secure positioning in wireless networks. IEEE Journal on Selected Areas in Communications: Special Issue on Security in Wireless Ad Hoc Networks **24**(2) (2006) 221–232
14. S. Čapkun, M.C., Srivastava, M.: Securing localization with hidden and mobile base stations. Internet-draft, NESL, UCLA (2005)
15. Krumm, J., Horvitz, E.: LOCADIO: Inferring motion and location from Wi-Fi signal strengths. In: First Annual Internationl Conference on Mobile and Ubiquitous Systems: Networking and Services, IEEE (2004) 4–13
16. Sastry, N., Shankar, U., Wagner, D.: Secure verification of location claims. [22] 1–10
17. Hancke, G.P., Kuhn, M.G.: An RFID distance bounding protocol. In: IEEE SecureComm 2005, Athens, Greece, 5–9 September 2005, IEEE Computer Society (2005) 67–73

18. R. Zetik, J.S., Thome, R.: UWB localization – active and passive approach. In: 21st IEEE Instrumentation and Measurement Technology Conference, IEEE (2004) 1005–1009
19. R.J. Fontana, E.R., Barney, J.: Commercialization of an ultra wideband precision asset location system. In: Conference on Ultra Wideband Systems and Technologies, IEEE (2003) 369–373
20. M. Ghavami, L.M., Kohno, R.: Ultra Wideband Signals and Systems in Communication Engineering. Wiley (2004)
21. Ubisense: White papers and datasheets. http://www.ubisense.net (2003–2006)
22. Maughan, W.D., Perrig, A., eds.: Proceedings of the 2003 ACM Workshop on Wireless Security, San Diego, CA, USA, September 19, 2003. In Maughan, W.D., Perrig, A., eds.: Workshop on Wireless Security, ACM (2003)
23. Crossbow Technology: MICA2 mote (2006) http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf.

A Distance Bounding with Existing Sensor Motes

Depending on the required spatial resolution, the communication requirements for a distance-bounding system can be quite stringent and are likely to exceed the capabilities of standard hardware. The MICA2 [23] mote, to name one illustrative example, has a communication rate of 38.4 kbit/s on its radio channel. In other words, a single bit lasts 26042 ns and is 7.8 km long. This means that the previously described attacks to shortcut the duration of a single bit with special hardware have the potential to manipulate a distance bound by several kilometers, many times the mote's nominal communication radius of 300 m. And this does not even take into account yet any protocol overhead (additional bits added at the start and end of a transmission frame) that the mote hardware relies on. Even if these constraints could be eliminated, the mote's 8 MHz clock still only permits its logic circuits to discriminate time intervals in 125 ns increments at best. In terms of a message round-trip, this still limits the distance resolution to at least 20 m.

For effective distance bounding, such a mote would have to implement a fast distance-bounding channel in addition to its slower standard communication channel. This separate distance-bounding channel would be optimized according to the principles listed in Section 3.6 towards the rapid turnaround exchange of single-bit messages, rather than for maximum range and reliability.